L Number	Hits	Search Text	DB	Time stamp
1	0	((interface adj device) same (stor$5 or memory) same port same serial$4 same (non-standard or non-application or ("not" near enabl$5)) same network)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:08
2	0	((interface adj device) same (stor$5 or memory) same port same serial$4 same (non-standard or non-application or ("not" near enabl$5)) same network) and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:08
3	0	((interface adj device) same (stor$5 or memory) same port same serial$4 same (non-standard or non-application or ("not" near support$5)) same network) and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:22
4	3	((interface adj device) same (stor$5 or memory) same port same serial$4 same (non-standard or non-application or ("not" near support$5) or standard) same network) and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:23
5	44	(serial near interface near device) and (network same manag$5) and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:24
6	14	(serial near interface near device) and (network same manag$5) and @ad<20001229 and (storage or memory) and ((network or serial) near port)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:25
7	4	(serial near interface near device) and (network same manag$5) and @ad<20001229 and (storage or memory) and ((network or serial) near port) and (non-standard same device)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:29
8	22	(interface near device) and (network same manag$5) and @ad<20001229 and (storage or memory) and ((network or serial) near port) and (non-standard same device)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:29
9	22	(interface near device) and (network same manag$5) and @ad<20001229 and (storage or memory) and ((network or serial) near port) and (non-standard same device)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:38
10	0	((interface with device) same (storage or memory) same (network adj port) same (serial adj port) same (processor or microprocessor)) and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:40
11	79	((interface with device) same (storage or memory) same (network with port) same (serial with port) same (processor or microprocessor)) and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:40
12	1	((interface with device) same (storage or memory) same (network with port) same (serial with port) same (processor or microprocessor)) and @ad<20001229 and (command or instruction) and (non-standard or ("not" near standard) or ("not" near enabled))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:43
13	57	((interface with device) same (storage or memory) same (network with port) same (serial with port) same (processor or microprocessor)) and @ad<20001229 and (command or instruction)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:43

14	1	((interface with device) same (storage or memory) same (network with port) same (serial with port) same (processor or microprocessor)) and @ad<20001229 and (command or instruction) and (non-standard or ("not" adj standard))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:45
15	0	((interface with device) same (storage or memory) same (network with port) same (serial with port) same (processor or microprocessor)) and @ad<20001229 and (command or instruction) and 709/220.ccls.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:46
16	0	((interface with device) same (storage or memory) same (network with port) same (serial with port) same (processor or microprocessor)) and @ad<20001229 and (command or instruction) and 709/221.ccls.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 18:07
-	39	(((network near device) or router or hub or switch) and cli).ab.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 15:06
-	39	(((network near device) or router or hub or switch) and cli).ab.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:04
-	0	(((network near device) or router or hub or switch) and cli).ab. and @ad<20001229 and command and instruction	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:04
-	1	(((network near device) or router or hub or switch) and cli).ab. and @ad<20001229 and (command or instruction) and interfac$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:04
-	2	(((network near device) or router or hub or switch) and cli).ab. and @ad<20001229 and (command or instruction)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:05
-	19	(((network near device) or router or hub or switch) and cli).ab. and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:06
-	3	(((network near device) or router or hub or switch) and ((command-line adj interface) or (command near line near interfac$5))).ab. and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:26
-	3	(((network near device) or router or hub or switch) and ((command-line adj interface) or (command near line near interfac$5))).clm. and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:13
-	48	(((network near device) or router or hub or switch) same ((command-line adj interface) or (command near line near interfac$5))).detd. and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:15
-	3	(((network near device) or router or hub or switch) same ((command-line adj interface) or (command near line near interfac$5)) same (program$5 with interfac$5)).detd. and @ad<20001229	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:25

-	1	(((network near device) or router or hub or switch) same ((command-line adj interface) or (command near line near interfac$5)) same (program$5 with interfac$5)).detd. and @ad<20001229 and java	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:25
-	0	(((network near device) or router or hub or switch) and ((command-line adj interface) or (command near line near interfac$5))).ab. and @ad<20001229 and java	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 10:26
-	2	6625590.pn. and (program$6 with interfac$5)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 11:34
-	1	6625590.pn. and (object or class)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 11:40
-	2	6625590.pn. and (method or call)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 11:45
-	1	6625590.pn. and (configur$5)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 11:46
-	1	6625590.pn. and (enabl$6 or power)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 11:46
-	0	(((network near device) or router or hub or switch) and cli).ab. and (remote$3 with serial$5)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/05/12 17:06

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore® RELEASE 1.7

Welcome
United States Patent and Trademark Office

» Sea

Help FAQ Terms IEEE Peer Review

Quick Links

Welcome to IEEE Xplore®

- ○- Home
- ○- What Can I Access?
- ○- Log-out

Tables of Contents

- ○- Journals & Magazines
- ○- Conference Proceedings
- ○- Standards

Search

- ○- By Author
- ○- Basic
- ○- Advanced

Member Services

- ○- Join IEEE
- ○- Establish IEEE Web Account
- ○- Access the IEEE Member Digital Library

Your search matched **0** of **1037503** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

cli<and>programming interface<and>network device Search

□ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

Results:
No documents matched your query.

Print Format

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search | Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting | No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

Welcome to IEEE Xplore®

- Home
- What Can I Access?
- Log-out

Tables of Contents

- Journals & Magazines
- Conference Proceedings
- Standards

Search

- By Author
- Basic
- Advanced

Member Services

- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

Print Format

Your search matched **913** of **1037503** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

cli<and>programming interface<and>router or hub or ad Search

□ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 **The END: a network adapter design tool**
Indiresan, A.; Mehra, A.; Shin, K.G.;
INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer an Communications Societies. Proceedings. IEEE , Volume: 2 , 29 March-2 April 1
Pages:756 - 764 vol.2

[Abstract] [PDF Full-Text (972 KB)] **IEEE CNF**

2 **Dynamic reconfiguration of hub-based networks**
Bisdikian, C.; Patel, B.; Schaffa, F.; Willebeek-LeMair, M.;
Systems Management, 1993., Proceedings of the IEEE First International Wor on , 14-16 April 1993
Pages:125 - 132

[Abstract] [PDF Full-Text (540 KB)] **IEEE CNF**

3 **A rate adaption coprocessor for terminal adapters with U-interface modems**
Rabaey, D.H.; Busschaert, H.J.; Reusens, P.P.; Verpooten, L.M.;
Solid-State Circuits Conference, 1989. Digest of Technical Papers. 36th ISSCC 1989 IEEE International , 15-17 Feb. 1989
Pages:152 - 153, 321

[Abstract] [PDF Full-Text (376 KB)] **IEEE CNF**

4 **Parametric detection of the helicopter hub echo**
Kulpa, K.; Czekala, Z.; Misiurewicz, J.; Falkiewicz, J.;
Radar Conference, 1999. The Record of the 1999 IEEE , 20-22 April 1999
Pages:262 - 266

[Abstract] [PDF Full-Text (384 KB)] **IEEE CNF**

5 **Transparent dissemination of adapters in Jini**
Vayssiere, J.;
Distributed Objects and Applications, 2001. DOA '01. Proceedings. 3rd International Symposium on , 17-20 Sept. 2001
Pages:95 - 104

[Abstract] [PDF Full-Text (768 KB)] **IEEE CNF**

6 **Improved performance token ring network interface adapter**
Radziejewski, I.R.; Lo, E.; Hardy, R.H.S.; Leung, A.M.;
Computers and Digital Techniques, IEE Proceedings E [see also Computers an Digital Techniques, IEE Proceedings-] , Volume: 137 , Issue: 6 , Nov. 1990
Pages:421 - 426

[Abstract] [PDF Full-Text (532 KB)] **IEE JNL**

7 **Efficient implementation of the SAR sublayer and the ATM layer in h speed broadband ISDN data terminal adapters**
Doumenis, G.; Reisis, D.I.; Stassinopoulos, G.I.;
Communications, 1993. ICC 93. Geneva. Technical Program, Conference Reco IEEE International Conference on , Volume: 1 , 23-26 May 1993
Pages:63 - 67 vol.1

[Abstract] [PDF Full-Text (480 KB)] **IEEE CNF**

8 **Comparison of methods for adapter characterization**
Randa, J.; Wiatr, W.; Billinger, R.L.;
Microwave Symposium Digest, 1999 IEEE MTT-S International , Volume: 4 , 1 June 1999
Pages:1881 - 1884 vol.4

[Abstract] [PDF Full-Text (272 KB)] **IEEE CNF**

9 **A high performance FDDI adapter for VAX systems**
Edgar, R.; Kalkunte, R.; Rege, S.; Russo, A.;
Compcon Spring '92. Thirty-Seventh IEEE Computer Society International Conference, Digest of Papers. , 24-28 Feb. 1992
Pages:265 - 268

[Abstract] [PDF Full-Text (220 KB)] **IEEE CNF**

10 **Using the MIL-C-38999 test fixture and an S-parameter test set to measure the transfer impedance of a new universal ground adapter**
Schade, N.F.; Obara, M.D.; Dixon, D.S.;
Electromagnetic Compatibility, 1990. Symposium Record. 1990 IEEE Internat Symposium on , 21-23 Aug. 1990
Pages:310 - 315

[Abstract] [PDF Full-Text (540 KB)] **IEEE CNF**

11 **A viable multi-standard rate adapter chip for ISDN communications**

Busschaert, H.J.; Reusens, P.P.; Van Camp, R.M.; Gouwy, C.M.;
Digital Communications, 1990. 'Electronic Circuits and Systems for Communications'. Proceedings., 1990 International Zurich Seminar on , 5-8 M 1990
Pages:231 - 242

[Abstract] [PDF Full-Text (236 KB)] **IEEE CNF**

---

12 **Proactive flight schedule evaluation at Delta Air Lines**
Schumacher, B.;
Simulation Conference Proceedings, 1999. Winter , Volume: 2 , 5-8 Dec. 1999
Pages:1232 - 1237 vol.2

[Abstract] [PDF Full-Text (696 KB)] **IEEE CNF**

---

13 **Gestalt VSAT service**
Robson, A.;
VSAT's in Europe - the Talking is Over, IEE Colloquium on , 9 Jun 1995
Pages:3/1 - 3/5

[Abstract] [PDF Full-Text (300 KB)] **IEE CNF**

---

14 **A rate adaption coprocessor for terminal adapters and U-interface modems**
Busschaert, H.J.; Reusens, P.P.; Rabaey, D.H.; Verpooten, L.M.;
Solid-State Circuits, IEEE Journal of , Volume: 24 , Issue: 6 , Dec. 1989
Pages:1625 - 1633

[Abstract] [PDF Full-Text (712 KB)] **IEEE JNL**

---

15 **Distributed component hub for reusable software components management**
Ci, J.X.; Wei-Tek Tsai;
Computer Software and Applications Conference, 2000. COMPSAC 2000. The Annual International , 25-27 Oct. 2000
Pages:429 - 435

[Abstract] [PDF Full-Text (568 KB)] **IEEE CNF**

---

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
25 26 27 28 29 30 31 32 33 34 Next

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search | Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting | No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

WIKIPEDIA
The Free Encyclopedia

Main Page | Recent changes | Edit this page | Page history
Printable version | Disclaimers

Not logged in
Log in | Help

Go Search

Other languages: Deutsch | Français | Magyar | 日本語 | Nederlands | Polski | Español | 中文(简体)

Main Page
Recent changes
Random page
Current events
Community Portal

Edit this page
Discuss this page
Page history
What links here
Related changes

Special pages
Contact us
Donations

Application Programming Interface

From Wikipedia, the free encyclopedia.

Server will be down for maintenance on 2004-05-12 from about 02:00 to 03:00 UTC.

An **Application Programming Interface** (API) is a set of definitions of the ways in which one piece of computer software communicates with another. It is a method of achieving abstraction, usually (but not necessarily) between lower-level and higher-level software.

One of the primary purposes of an API is to provide a set of commonly-used functions—for example, to draw windows or icons on the screen. Programmers can then take advantage of the API by making use of its functionality, saving them the task of programming everything from scratch. APIs themselves are abstract: software that provides a certain API is often called the *implementation* of that API.

For example, one may look at the task of writing "Hello World" on a screen at increasing levels of abstraction:

1. Do it all yourself:
   1. Draw, on graph paper, the shapes of the letters H, e, l, l, o, W, o, r, l, d.
   2. Work out a matrix of black and white squares that look like those letters.
   3. Devise a way to program the CPU to put this matrix into the display adapter's frame buffer.
   4. Set the graphics card up to scan its frame buffer to generate the correct signal.
2. Use an operating system to do some of the work:
   1. Load a data structure called a "font" provided with the operating system.
   2. Have the operating system display a blank window.
   3. Have the operating system draw the text "Hello World" to the window with the font.
3. Use an application program (which uses an operating system) to do all this hard work:
   1. Write an HTML document containing the text "Hello World".
   2. Open the document in a Web browser such as Mozilla or Internet Explorer.

Obviously, the first option requires more steps, each of them much more complicated than the steps in the subsequent options. In addition, it is impractical to use the first approach to present a substantial amount of information, such as an encyclopedia article, on the screen, whereas the second approach makes the task easier by an order, and the third approach basically requires only that you type "Hello World." However, higher-level APIs often lose flexibility; for example, it would be much more difficult in a web browser to rotate text around a point with blinking outlines, something that could be done easily at the lower levels. This difference is a typical example of the trade-offs encountered by choosing to use an API.

APIs are as essential to computers as electrical standards are to the home. One can plug his toaster into the wall whether he is at home or at a neighbour's house, because both houses conform to the standard API for an electrical socket. If there wasn't an API standard, one would have to bring a power station along to make toast! Note that there is nothing stopping somebody else coming up with another standard; a European toaster will not work in the US without a transformer, just as a program written for Microsoft Windows will not work directly on a UNIX system without an intermediate API adapter such as WINE.

There are various design models for APIs. Interfaces intended for the fastest execution often consist of sets of functions, procedures, variables and data structures. However, other models exist as well, such as the interpreter used to evaluate expressions in ECMAScript/JavaScript. A good API provides a "black box" or abstraction layer, which prevents the programmer from needing to know how the functions of the API relate to the lower levels of abstraction. This makes it possible to redesign or improve the functions within the API without breaking code that relies on it.

Two general lines of policies exist regarding publishing APIs:

1. Some companies guard their APIs zealously. For example, Sony makes its official PlayStation API available only to licensed PlayStation developers. This is because Sony wants to restrict how many people can write a PlayStation game, and wants to profit from them as much as possible. This is typical of companies who do not profit from the sale of API implementations (in this case, Sony breaks even by selling PlayStation consoles and takes a loss on marketing, intending to make it up through game royalties created by API licensing).
2. Other companies propagate their APIs freely. For example, Microsoft deliberately makes its API information public, so that software will be written for the Windows platform. The sale of the third-party software sells copies of Windows. This is typical of companies who profit from the sale of API implementations (in this case, Microsoft Windows, which is sold at a gain for Microsoft).

Some APIs, such as the ones standard to an operating system, are implemented as separate code libraries that are distributed with the operating system. Others require software publishers to integrate the API functionality directly into the application. This forms another distinction in the examples above. Microsoft Windows APIs come with the operating system for anyone to use. Software for embedded systems such as video game consoles generally falls into the application-integrated category. While an official Playstation API document may be interesting to read, it is of little use without its corresponding implementation, in the form of a separate library or developer's kit.

An API that does not require royalties for access and usage is called "open." The APIs provided by Free software (such as software distributed under the GNU General Public License), are open by definition, since anyone can look into the source of the software and figure out the API. Although usually authoritative "reference implementations" exist for an API (such as Microsoft Windows for the Win32 API), there's nothing that prevents the creation of additional implementations. For example, most of the Win32 API can be provided under a UNIX system using software called WINE.

It is generally lawful to analyze API implementations in order to produce a compatible one. This technique is called reverse engineering for the purposes of interoperability. However, the legal situation is often ambiguous, so that care and legal counsel should be taken before the reverse engineering is carried out. For example, while APIs usually do not have an obvious legal status, they might include patents that may not be used until the patent holder gives permission.

Example APIs

- The PC BIOS call interface
- Document Object Models of various applications, such as HTML
- The Single UNIX Specification
- The Microsoft Win32 API
- The J2EE Java 2 Enterprise Edition APIs
- ASPI for SCSI device interfacing
- The Carbon and Cocoa APIs for the Macintosh OS
- Simple Network Management Protocol (SNMP)
- Universal Plug and Play (UPNP)
- Common Object Request Broker Architecture (CORBA)
- The various APIs which make up Microsoft's DirectX.
- Simple DirectMedia Layer (SDL)

See also:

- Ontology (computer science)

Edit this page | Discuss this page | Page history | What links here | Related changes

Other languages: Deutsch | Français | Magyar | 日本語 | Nederlands | Polski | Español | 中文(简体)

Main Page | About Wikipedia | Recent changes | □

Go Search

FOLDOC Free On-Line Dictionary Of Computing
Application Programming Search Home Contents Feedback Random

Application Programming Interface ==>

Application Program Interface

<*programming*> (API, or "application programming interface") The interface (calling conventions) by which an application program accesses operating system and other services. An API is defined at source code level and provides a level of abstraction between the application and the kernel (or other privileged utilities) to ensure the portability of the code.

An API can also provide an interface between a high level language and lower level utilities and services which were written without consideration for the calling conventions supported by compiled languages. In this case, the API's main task may be the translation of parameter lists from one format to another and the interpretation of call-by-value and call-by-reference arguments in one or both directions.

(1995-02-15)

Try this search on OneLook / Google

Nearby terms: application layer « Application Portability Architecture « application program « **Application Program Interface** » Application Programming Interface » Application Protocol Data Unit » Applications Development Manager